



**Actividad 03**

**Alumno: José Osvaldo Farfán de León**

**Materia: Seminario de IA II**

**Profesor: Carlos Alberto Villaseñor Padilla**

**Sección: "D04"**

## Explicación de variables con regresión lineal múltiple

- Descarga el dataset de "home\_data.csv"
- Limpia y escala los datos
- Crea una regresión lineal múltiple sobre la variable del precio y todas las otras variables como entrada
- Extrae los coeficientes del modelo
- Gráfica en barras los coeficientes del modelo en orden
- Explica porque crees que se obtuvieron esos resultados
- Crea un modelo de LASSO ( Least Absolute Shrinkage and Selection Operator)
- Extrae los coeficientes del modelo
- Gráfica en barras los coeficientes del modelo en orden
- Crea un documento PDF con tus experimentos

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression, Lasso
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import mean_squared_error

# Cargar el conjunto de datos
data = pd.read_csv("home_data.csv")

# Limpieza y escala de los datos
data.dropna(inplace=True)

# Dividir el conjunto de datos en características (X) y variable objetivo (y)
X = data.drop("price", axis=1)
y = data["price"]

# Escalar las características
scaler = StandardScaler()
X = scaler.fit_transform(X)

# Dividir los datos en conjuntos de entrenamiento y prueba
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Regresión lineal múltiple
regression_model = LinearRegression()
regression_model.fit(X_train, y_train)
regression_coefs = regression_model.coef_

# Modelo de LASSO
lasso_model = Lasso(alpha=1.0) # Puedes ajustar el valor de alpha
lasso_model.fit(X_train, y_train)
lasso_coefs = lasso_model.coef_
```

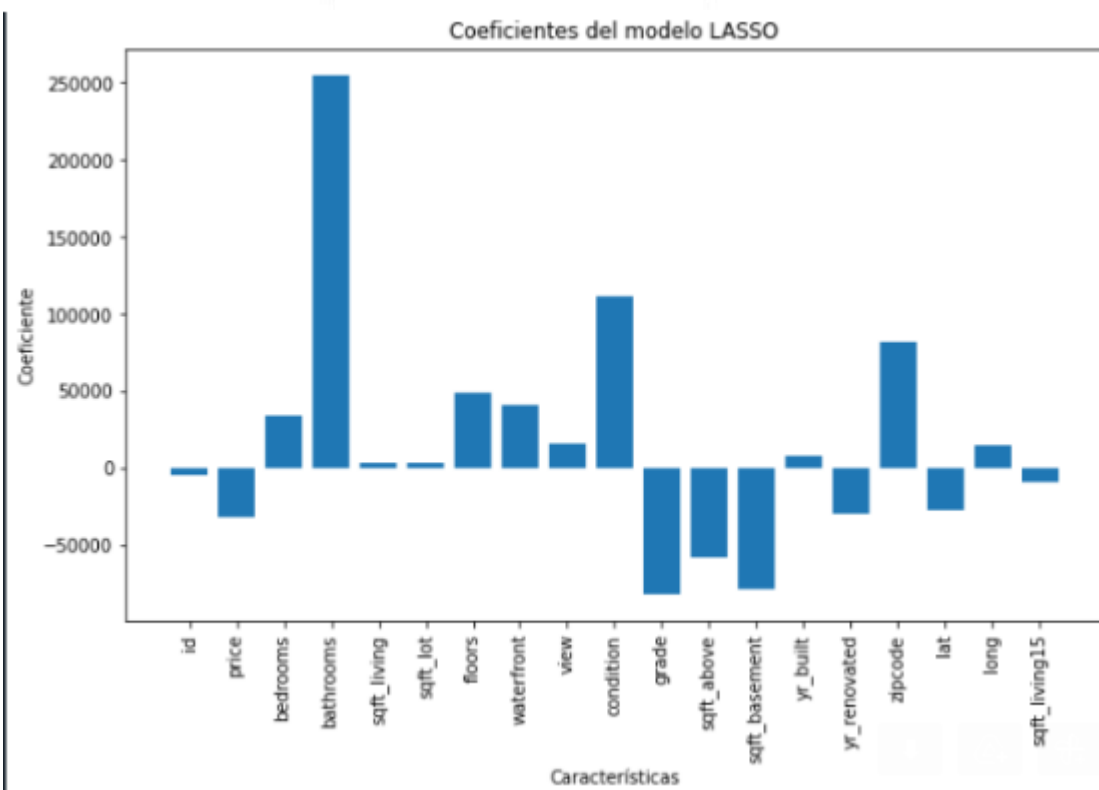
```
# Gráfica de barras de los coeficientes del modelo de regresión lineal
plt.figure(figsize=(10, 6))
plt.bar(range(len(regression_coefs)), regression_coefs)
plt.xticks(range(len(regression_coefs)), data.columns[:-1], rotation=90)
plt.title("Coeficientes del modelo de regresión lineal")
plt.xlabel("Características")
plt.ylabel("Coeficiente")
plt.show()

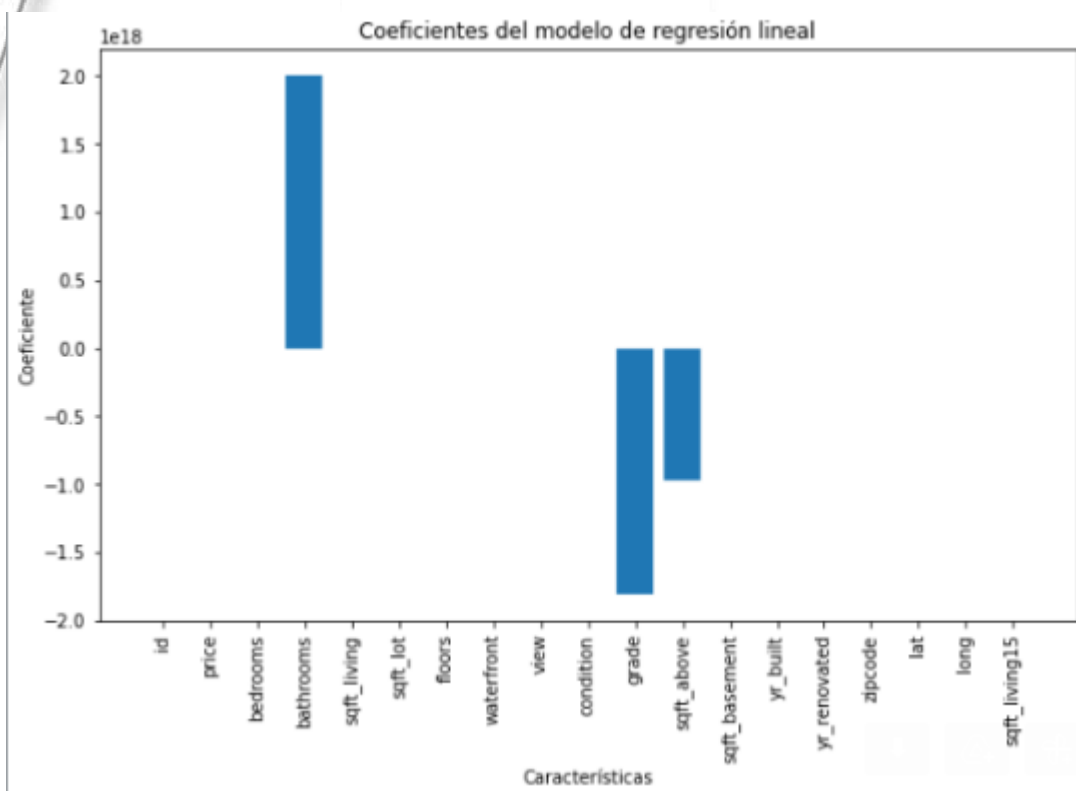
# Gráfica de barras de los coeficientes del modelo LASSO
plt.figure(figsize=(10, 6))
plt.bar(range(len(lasso_coefs)), lasso_coefs)
plt.xticks(range(len(lasso_coefs)), data.columns[:-1], rotation=90)
plt.title("Coeficientes del modelo LASSO")
plt.xlabel("Características")
plt.ylabel("Coeficiente")
plt.show()

# Calcular el error cuadrático medio para ambos modelos
regression_predictions = regression_model.predict(X_test)
lasso_predictions = lasso_model.predict(X_test)

mse_regression = mean_squared_error(y_test, regression_predictions)
mse_lasso = mean_squared_error(y_test, lasso_predictions)

print(f"MSE del modelo de regresión lineal: {mse_regression}")
print(f"MSE del modelo LASSO: {mse_lasso}")
```





MSE del modelo de regresión lineal: 45166331069.9702  
MSE del modelo LASSO: 45164857848.17046